

FIG. 1A 29 Basic Strokes:

丶	一	丨	ノ	㇏	㇏	㇏	㇏
㇏	㇏	㇏	㇏	㇏	㇏	㇏	㇏
㇏	㇏	㇏	㇏	㇏	㇏	㇏	㇏
㇏	㇏	㇏	㇏	㇏			

8 Stroke Categories:

一	丨	ノ	丶	㇏	㇏	㇏	㇏
---	---	---	---	---	---	---	---

5 Stroke Categories:

一	丨	ノ	丶	㇏
---	---	---	---	---

FIG. 1B 29 to 8 classification

一 ㇏	一
丨 ㇏	丨
ノ	ノ
丶 ㇏	丶
㇏ ㇏ ㇏ ㇏ ㇏ ㇏ ㇏	㇏
㇏ ㇏ ㇏ ㇏ ㇏ ㇏ ㇏	㇏
㇏ ㇏ ㇏	
㇏	㇏
㇏ ㇏ ㇏	㇏

FIG. 1C 29 to 5 classification

一 ㇏	一
丨 ㇏	丨
ノ	ノ
丶 ㇏	丶
㇏ ㇏ ㇏ ㇏ ㇏ ㇏ ㇏ ㇏	㇏
㇏ ㇏ ㇏ ㇏ ㇏ ㇏ ㇏ ㇏	
㇏ ㇏ ㇏ ㇏ ㇏ ㇏ ㇏	

SECRETED 6322050

字	国	部	件	笔顺	笔	划
啊	B0A1	口	可	10	丨	㇏
阿	B0A2	阝	可	7	㇏	丨
埃	B0A3	扌	厶 矢	10	一	丨
挨	B0A4	扌	厶 矢	10	一	丨
哎	B0A5	口	ㄗ ㄨ	8	丨	㇏
唉	B0A6	口	厶 矢	10	丨	㇏
哀	B0A7	宀	口 依	9	丶	一
皑	B0A8	白	山 己	11	ノ	丨

2.1 Character

2.2 Component composition

2.3 Stroke composition

FIG. 2

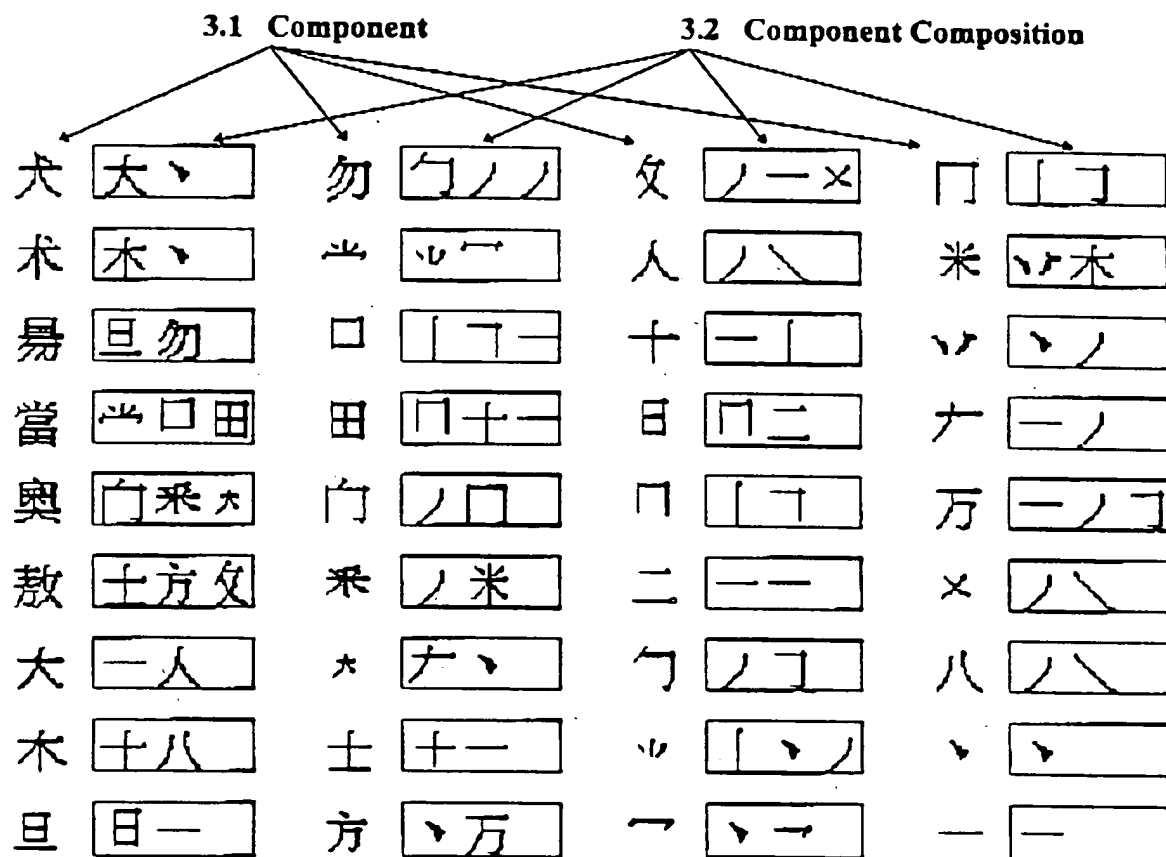


FIG. 3

SECRET 000000

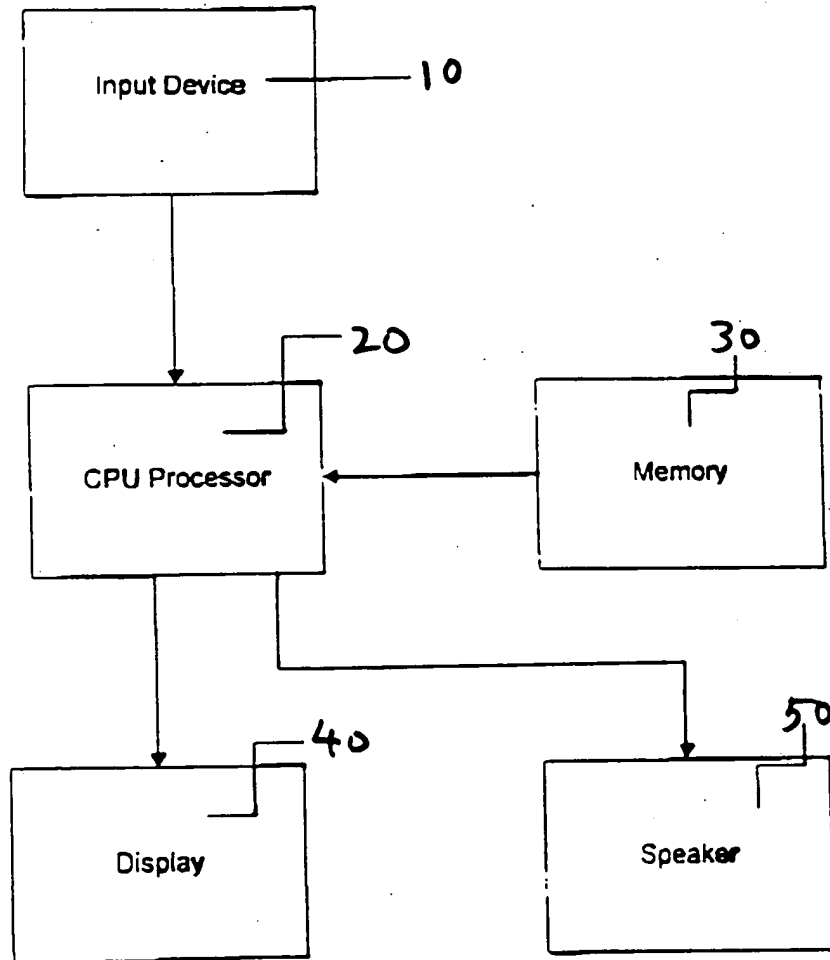


FIG. 4

65220-6522000

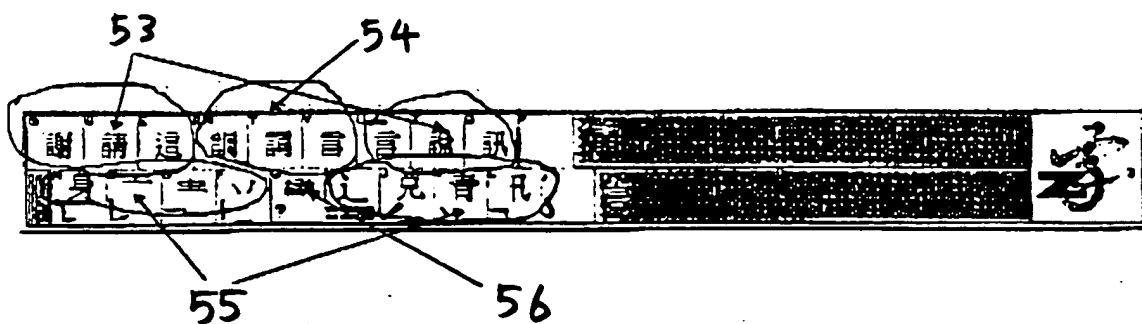
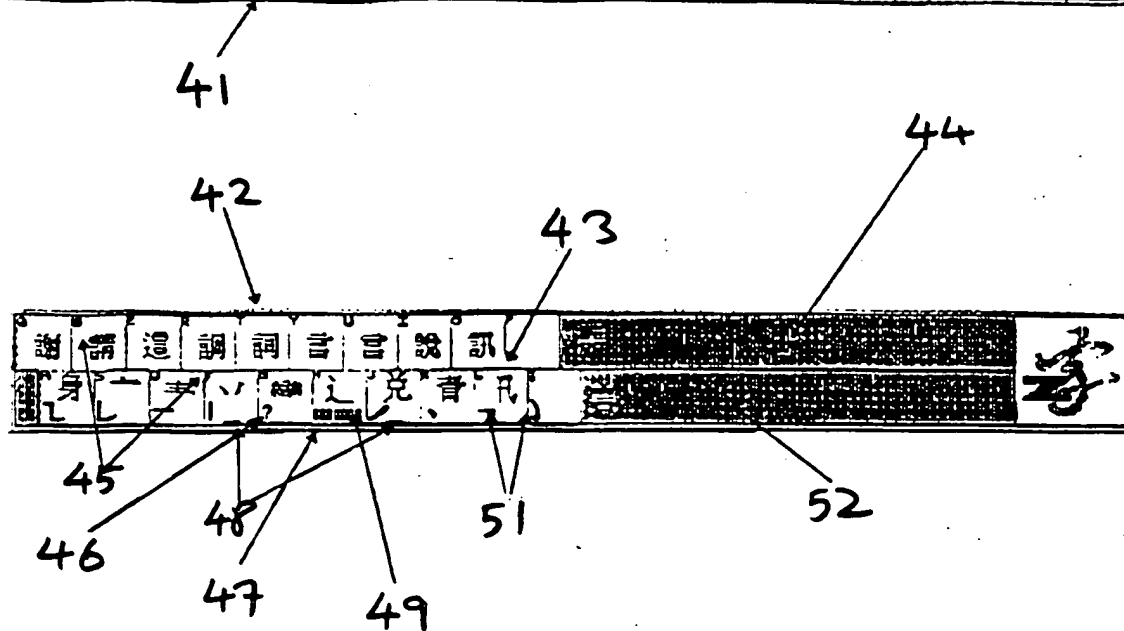
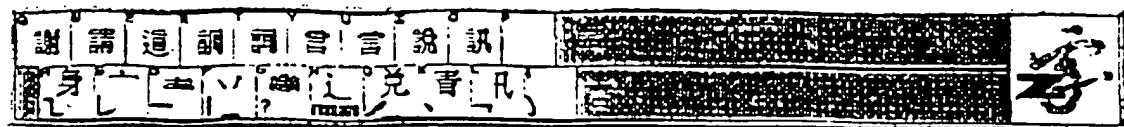
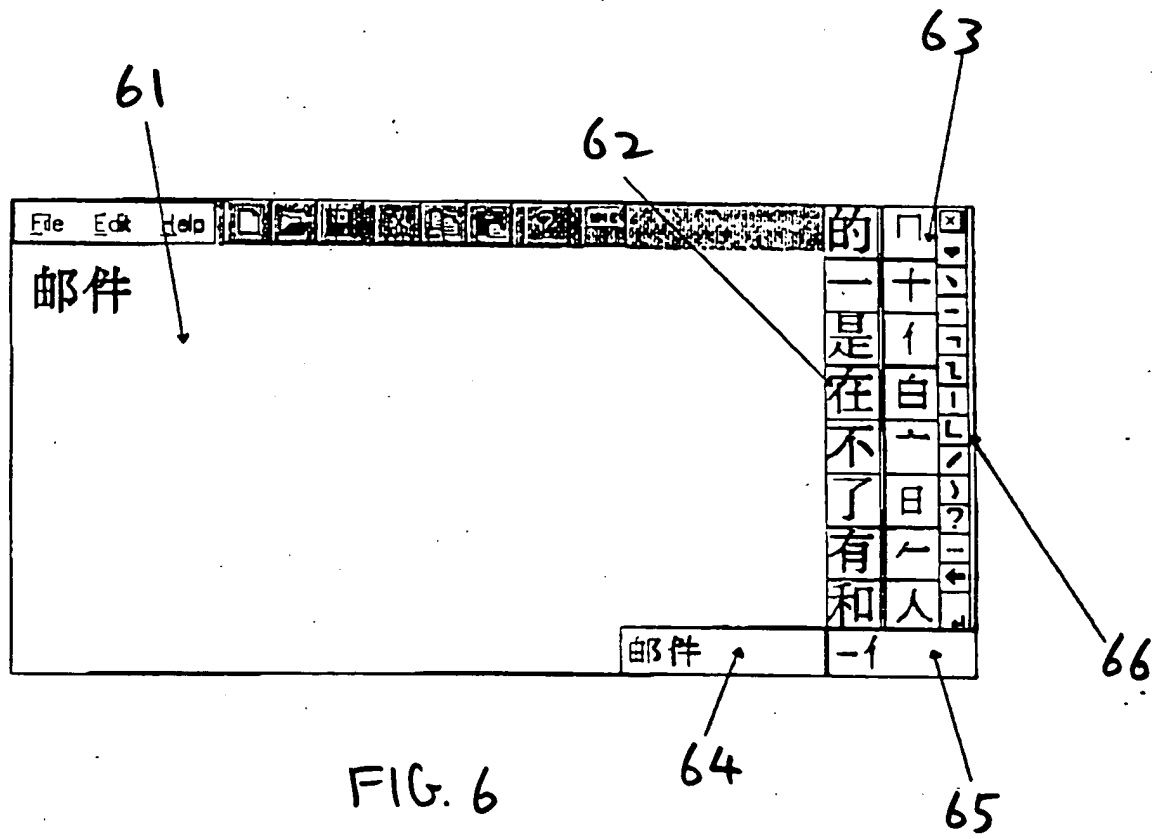


FIG. 5



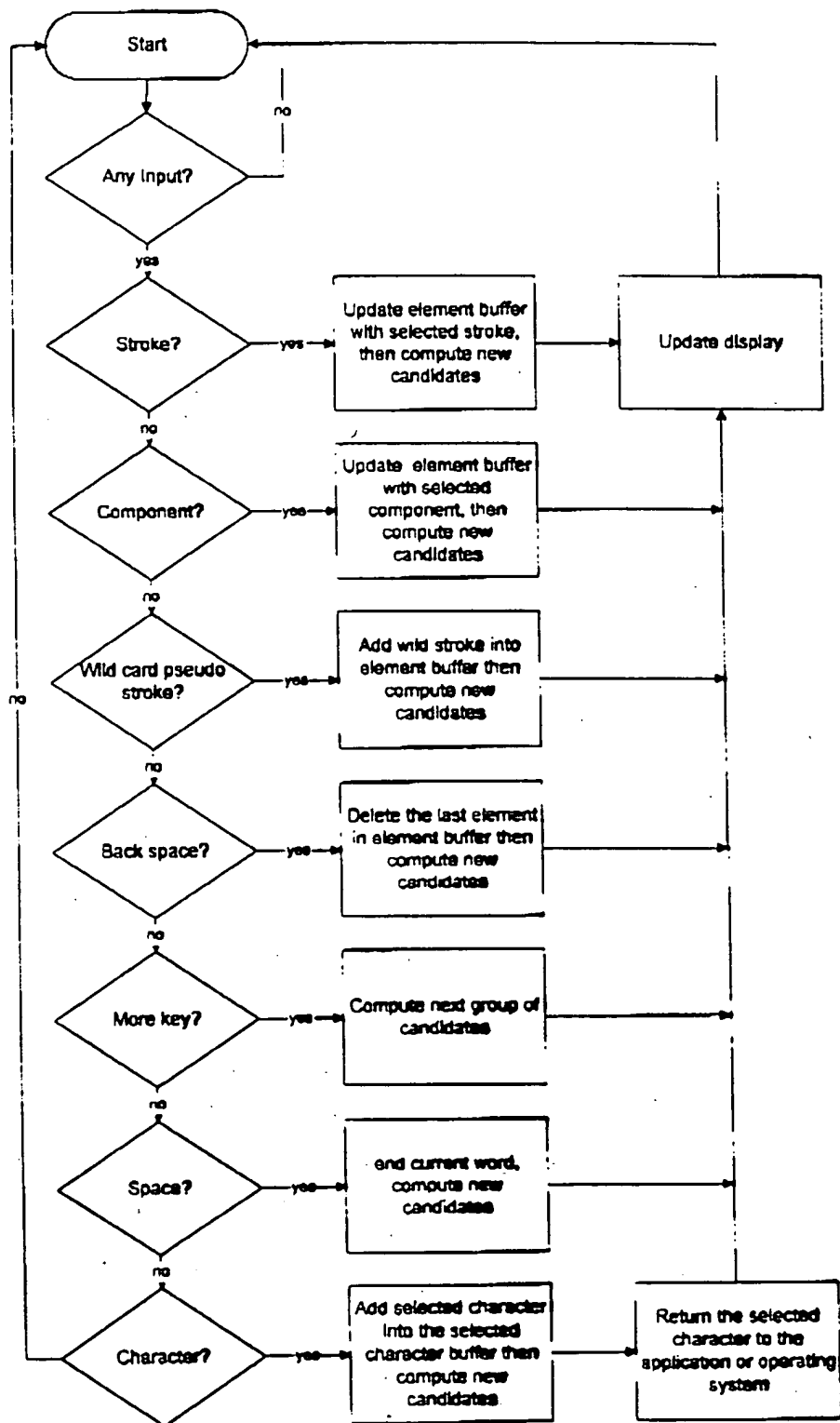
[illegible]

FIG. 7

8.1 Character_Table: array of structure CharInfo

Structure CharInfo:

8.2 UCHAR ucStrokes[12]; // Stroke composition Info
// 24 strokes stored as nybbles

8.3 ULONG ulComponents; // 10 bit first drawn component
// 10 bit second drawn component
// 8 bit KangXi radical
// 4 bits of properties

8.4 USHORT usFreq; // frequency of the character

8.5 USHORT usCode; // base codeset code of the character

8.6 wchar_t wsInitial[3]; // the initial sound of the character

8.7 wchar_t wszFinal[7]; // the final sound of the character

8.7 ULONG ulOffsetWords; // pointer to words beginning with this char

8.8 Component Table: array of structure Component

Structure Component:

8.9 UCHAR ucStrokes[12]; // Stroke composition Info

8.10 ULONG ulComponent; // the first drawn component

8.11 Word Table: array of structure Word

Structure Word:

array of ULONG word(); // code of the words

8.12 ULONG ulWordFreq; // high bit is set + Word Freq.

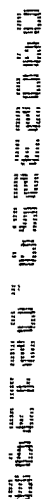
8.13 Component Frequency Table:
array of ULONG ulFreq; // size in: total number of components

8.14 Complex Component Frequency Table:
array of ULONG ulFreq; // size in: total number of components

8.15 Stroke Match Buffer: UCHAR ucStrokes[12];

8.16 Letter Buffer: array of char cLetters[];

FIG. 8



人 大 一 不 的 是 我 有 在

論語集注 卷之四 子罕篇 子罕見之 子罕見之

[illegible]

氣系靈宮句詞云文辨
在日自仁一十口言

謝開遺訓詞言言說錄
身一二三，學上兌青凡

在有我是學的不一大

FIG. 9

```
graph TD; Start(( )) --> 10.1[Initialization Process]; 10.1 --> 10.2[Expand element buffer]; 10.2 --> 10.3[Matching with Character Table]; 10.3 --> 10.5{Done?}; 10.5 -- yes --> 10.4[Matching with Word Table]; 10.5 -- no --> 10.3; 10.4 --> 10.6{Done?}; 10.6 -- yes --> 10.7[Display Process]; 10.6 -- no --> 10.4; 10.7 --> End(( ))
```

The flowchart illustrates the character matching process, starting with an initialization process (10.1), followed by expanding the element buffer (10.2). It then enters a loop for matching with the character table (10.3). If this step is done, it proceeds to matching with the word table (10.4). If this step is also done, it proceeds to the display process (10.7). If either step is not done, it loops back to the previous matching step.